基于复合特征和动态阈值圆法的手势识别算法研究 *

王　梅，张　震，张　曦，屠大维

(上海大学 机电工程与自动化学院，上海 200072)

摘　要：针对传统手势识别中用肤色分割手部区域效果的局限性，采用 Kinect 获取深度信息来分割手掌，能得到较好效果。对手掌轮廓进行多边形逼近，将凸包点作为候选指尖点。利用非零像素（白）到最近零像素的距离提取掌心，用线性回归动态调整阈值圆半径，将无用凸包点过滤，实现指尖点准确提取。在分类识别中，将图像的 Hu 矩和指尖点个数组合起来，作为复合手势特征，导入 KNN 分类器中，实现手势识别。实验证明，基于复合特征和动态阈值圆法的手势识别算法具有较好的识别率和实时性。

关键词：手势识别；几何不变矩；动态阈值圆；KNN

中图分类号：TP391.41　　doi: 10.19734/j.issn.1001-3695.2018.04.0491

Gesturerecognitionthat based on combined features and dynamic threshold circle

Wang Mei, Zhang Zhen, Zhang Xi, Tu Dawei

(*College of Mechatronics Engineering & Automation*, *Shanghai University*, *Shanghai* 200072, *China*)

Abstract: For the limitation of hand region segmention with skin color in the traditional gesture recognition, using Kinect to obtaining depth information to segment the palm can get better results. The palm contour is polygonal approximation, and the convex points are selected as the candidate fingertip. The fingertips are distinguished accurately from the unwanted convex points by a threshold circle. Utilizing the distance relation of the non zero pixel (white) to the nearest zero pixel to extract the palm center, the threshold circle transform radius dynamically by linear regression method. In the classification and recognition, combining the Hu moment and the number of fingertip as the feature of gesture, and the KNN classifier achieve gesture recognition. Experiments show that the gesture recognition algorithm based on composite features and dynamic threshold circle method has better recognition rate and real-time performance.

Key words: gesture recognition; geometric invariant moment; dynamic threshold circle; KNN

0　引言

作为新一代人机交互的重要手段，手势识别成为近年来的研究热点，在虚拟现实、远程控制和交通等领域得到广泛应用。早期研究集中于专用设备的研制，如数据手套可以提供手部各关节数据[1]，用于精确手势识别，但设备昂贵、过程繁琐制约了其应用；其后，通过在手部做标记的标记手势研究[2]发展起来，但受限于标志点位置和数量等因素，识别手势种类有限，局限性较大。因而，基于视觉的手势识别方法[3]应运而生。

基于视觉的手势识别系统主要分为基于 RGB 图像和基于深度图像两类。传统的 RGB 图像的手势分割在 HSV（或 YCrCb）颜色空间下，通过肤色与背景的明显差异，将手部分离出来[4]。该方法对光线、场景等要求较高，且当人体其他部位（如脸）出现时，会造成较大干扰，效果较差。在此基础上，研究者又提出了利用灰度直方图[5]来进行手掌检测，灰度直方图依赖于图像灰度的阶跃变化，能够较好的分割手势，但是该过程计算较为复杂，需要花费较多时间时间。与 RGB 图像相比，深度图像具有物体三维特征信息，即深度信息。由于深度图像不受光源照射方向及物体表面的发射特性的影响，而且不存在阴影，不受颜色相近信号影响，所以可以更准确迅速地识别手势[6]。

基于视觉的手势识别需要经过分割、表示、识别等三个阶段[7]。手势分割阶段主要是将手掌从环境中分割出，在该环节中主要解决背景对手势分割的影响。手势表示则通过一定的特征来表征人手模型，较常用的是以指尖特征和掌心特征为重要环节的轮廓特征[8]。文献[9]中将手掌分区并对不同区域设置权值来确定掌心，减少手指部分对掌心位置的影响；此方法虽可获取较好的掌心位置，但是对区域划分要求较高；文中采用基于轮廓曲率的指尖点识别方法。高晨等人[10]采用计算手掌轮廓

收稿日期：2018-04-27；**修回日期：**2018-06-05　　**基金项目：**国家重点研发计划重点专项课题（2016YFC0302402）

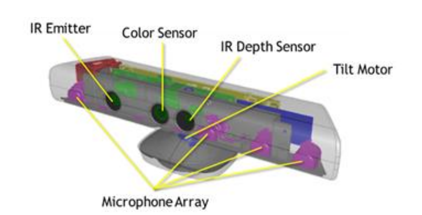
作者简介：王梅（1969-），女，辽宁人，硕导，博士，主要研究方向为机器人学、机器视觉及伺服控制、智能系统等；张震（1993-），男，湖北人，硕士研究生，主要研究方向为机器人学、模式识别等；张曦（1977-），男（通信作者），江苏人，副教授，硕导，博士，主要研究方向为视觉检测技术（xizhang@shu.edu.cn）；屠大维（1965-），男，浙江人，教授，博导，主要研究方向为机器视觉及伺服控制、光机电一体化智能机械及仪器、人机智能系统等.

最大内切圆的方向确定掌心位置，对于特定手势适用性有限。手势识别以特征为依据，判别人体手势含义，常用的方法有基于模板匹配法与分类算法[11-13]。

本文提出一种基于复合特征和动态阈值圆法的手势识别算法，将分别从手势分割、手势特征提取、手势分类识别等方面加以介绍。

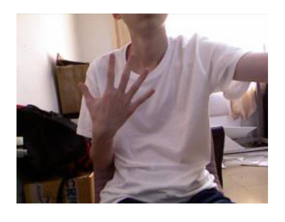
1 手势分割

本文利用 Kinect 的彩色相机和深度传感器，获取人体信息（图 1），以完成手势识别。



(a)Kinect 相机

(a)Kinect camera





(b)彩色图像 (c)深度图像

(b) color image (c) depth image

图 1 Kinect 相机及其获取的彩色图像与深度图像

Fig.1 Kinect camera and its acquired and depth image

利用 Kinect 和 OpenNI 的手掌跟踪算法获取手掌心的三维坐标（x_H, y_H, z_H）。利用场景中其他杂体和手掌与传感器的距离（深度）差异，以深度信息 z_H 为标准，合理设置阈值 a，可以将手掌从场景中分割出来。

由于原始的深度图像包含过多的前景和背景信息，需要对其设置感兴趣区域（Region Of Interest，ROI），即手掌所在区域，以减少过多无关信息，提高程序运行速度。根据手掌心坐标中的 x_H、y_H，设置感兴趣区域矩形范围：$x \in [x_H - offset, x_H - offset]$、$y \in [y_H - offset, y_H - offset]$。根据实际手掌大小和 Kinect 的检测范围设置 offset=150。

在对手掌进行分割同时进行二值化处理，二值化的的操作同样是减少无关信息，提高图像处理速度。在本文的手掌分割中，以手掌的深度为阈值标准，手掌部分像素值设置为 255（深度介于 $[z_H - a, z_H + a]$ 之间），将手掌以外的背景信息的像素值设置为 0（黑）。

$$H(x,y)=\begin{cases}255, & D(x,y)\in[z_H-a,z_H+a]\\ 0, & D(x,y)=else\end{cases}$$,其中 $x \in [x_H - offset, x_H - offset]$,$y \in [\ y_H - offset, y_H - offset\ ]$ (1)

公式(1)中的阈值 a 取 15mm，$D(x,y)$为坐标为图像坐标为（x,y）处点的深度值，$H(x,y)$为坐标为（x,y）的点在分割并二值化后的图像中的灰度值，图像的大小为 ROI 所限定的范围。

经过上述操作获得如图 2 所示的清晰完整的手掌图像。



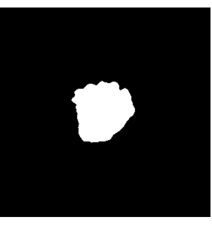




图 2 不同手势的深度分割图

Fig.2 Depth split diagram of different gestures

进而，采用中值滤波算法对分割出的手部图像进行滑动滤波，迭代过程中，当前窗口内的所有像素排序，保留中间值作为当前像素值。中值滤波的输出为

$$g(x,y)=med\{f(x-k,y-l)\} \tag{2}$$

其中：$f(x,y), g(x,y)$ 分别为原始图像和处理后的图像。

中值滤波方法具有较好地抑制噪声效果。经过滤波后的图像再进行二值化，得到的图形较稳定，可以提高后续图像处理的速度。

2 手势特征提取

在手掌分割的基础上，通过对手势特征的分类，完成手势识别。在众多手势特征中，本文选择图像几何不变矩的四个特征和指尖点特征，共同组成复合的手势特征，作为手势识别的依据，以避免单特征误差造成的识别结果波动。

2.1 几何不变矩特征提取

图像的几何矩是 1962 年由 Hu 提出的[14]，选择出来的 7 个几何矩不变量经过图像旋转、平移、缩放等操作后，保持不变性。因此在图像处理中，该属性通常作为重要特征用于分类等操作。本文选择其中的 4 个几何矩不变量，利用其不变性，作为手势特征的一个组成部分，用于手势识别。

对于图像 $f(x,y)$，其 $(p+q)$ 阶普通矩的公式为：

$$m_{pq}=\iint x^p y^q f(x,y)dxdy\ (p=0,1,2\ldots;q=0,1,2\ldots) \tag{3}$$

0 阶矩 m_{00} 表示目标的“质量”，1 阶矩（m_{10}, m_{01}）表示目标质心，二阶矩（m_{20}, m_{02}, m_{11}）表示旋转半径，三阶矩（$m_{30}, m_{03}, m_{12}, m_{21}$）描述目标的方位和斜度。

当图像连续时，利用质心 m_{01}, m_{10} 可得到中心矩：

$$\mu=\iint x^p-\frac{m_{10}}{m_{00}}\left(y^q-\frac{m_{01}}{m_{00}}\right)f(x,y)dxdy \tag{4}$$
$$p=0,1,2,\cdots;q=0,1,2,\cdots$$

为了消除尺度变化对其影响，通常将各阶中心矩通过零阶中心矩进行归一化，得到：

$$\eta=\frac{\mu_{pq}}{\mu_{00}^r} \tag{5}$$

其中 $r=\frac{p+q}{2}+1$。

根据 Hu 矩理论，得到 7 个不变矩特征量。由于高阶矩计算繁琐，对噪声更敏感[15]，本文取前四个作为分类特征：

$$\begin{cases} I_1 = \eta_{20} + \eta_{02} \\ I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}{}^2 \\ I_3 = (\eta_{30} - \eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ I_4 = (\eta_{30} + \eta_{12})^2 + (3\eta_{21} + \eta_{03})^2 \end{cases} \tag{6}$$

由于图像的几何不变矩之间的数量级相差较大，并且可能出现负数，为了方便后续操作，采用不变矩的自然对数对数据范围进行压缩。令

$$I_i^{'} = \ln(\ |I_i|\ ) \tag{7}$$

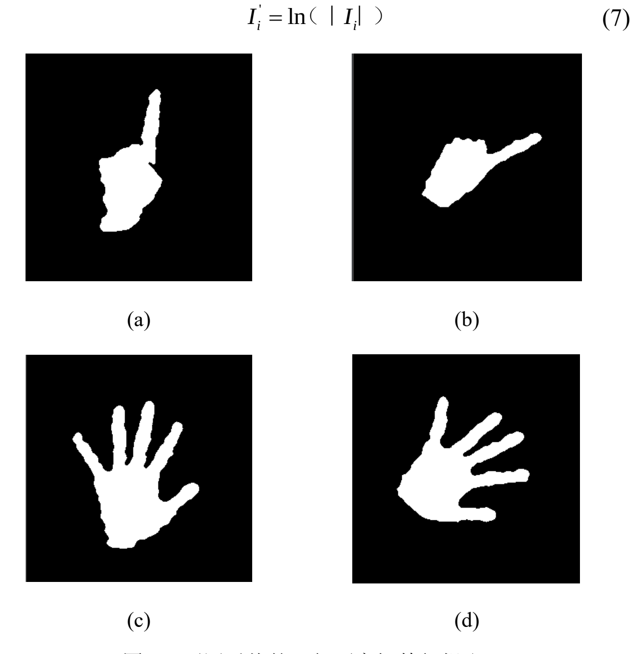


图 3　不同手势的几何不变矩特征提取

Fig.3　Feature extraction of geometric invariant moment for different gestures

根据图 3 的手势提取几何不变矩特征，得到表 1。

表 1　不同手势的几何不变矩特征提取

Table 1　Feature extraction of geometric invariant moment for different gestures

手势编号	$I_1^{'}$	$I_2^{'}$	$I_3^{'}$	$I_4^{'}$
1	-6.80197	-14.2192	-21.7712	-22.1432
2	-6.47943	-14.067	-20.6139	-20.8318
3	-6.91585	-16.3909	-26.6766	-23.5799
4	-6.98353	-16.9139	-25.7383	-24.1375

从表中可以看出，手势 1 和 2、手势 3 和 4 的几何不变矩特征分别相近，手势的方向及大小不会影响该特征。

2.2　指尖点检测

手势通常通过手指变换来表现的，通过指尖的几何关系，可以反映当前手势。在手势识别中，指尖信息是十分重要的。因而，本文选择指尖点信息，作为手势特征的另一个组成部分，与几何不变矩的前四个特征量一起，用于手势识别。

在指尖点检测过程中，首先基于 Douglas-Peucker 算法进行手轮廓多边形提取，然后基于凸包算法计算候选指尖点，最后通过阈值圆分类法完成指尖点检测。其中，阈值动态调整算法可以保证指尖提取的可靠性。

2.2.1 手的轮廓多边形提取

为了减少轮廓数据量，加快凸包计算和阈值圆分类的运行速度，在确定指尖点之前，根据获取的手掌轮廓图，提取多边形，以逼近手掌轮廓。

手的轮廓多边形提取采用 Douglas-Peucker 算法，该算法主要用于曲线近似表示。算法如下：

a)在曲线首尾（A,B）连接一条直线，计算曲线上其余各点到直线的距离；

b)选择距离的最大值，与预先给定的阈值比较；如果距离最大值小于阈值时，将该直线 AB 作为曲线的近似，处理完毕。

c)如果距离最大值大于阈值，以最大值点 C 为分割点，连结 AC、BC，并对这两段线重复 1、2 两部分的操作，重复迭代，直到曲线上所有的点到对应折线的距离均小于设定阈值。

d)处理完成后，连结各个分割点，形成的折线即为获得的近似曲线。

如图 4 所示，通过若干次迭代，依次搜索出分割点 C、D、E，曲线 AB 可近似表示为折线 AEDCB。

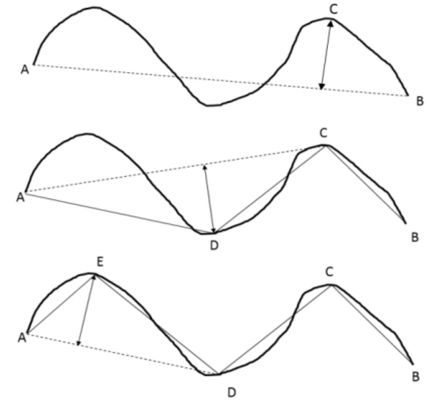


图 4 近似曲线提取

Fig.4　Approximate curve extraction byDouglas-Peucker algorithm

利用 Douglas-Peucker 算法，在手掌分割图的基础上，设置合适的阈值，得到图 5 中粗线条所示的几何多边形，较好地逼近手掌轮廓。

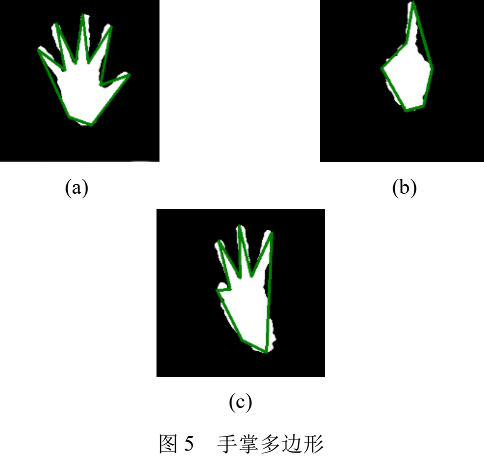


图 5　手掌多边形

Fig.5　Palm polygon

2.2.2 候选指尖点提取

利用凸包算法，可以得到候选指尖点。图 5(a)所示手势有 7 个候选指尖点。基于凸包算法的指尖点筛选法，能够有效提取指尖点，避免指根点等缺陷点的误提取。但是，对于手掌轮廓边缘与小臂交界处的点、手指弯曲时残留顶点等干扰点，难以区分，因而手势轮廓多边形的凸包点并非都是指尖点，需要进一步过滤这些无关点，阈值圆法可以解决这一问题。

2.2.3 阈值圆指尖点提取

在某一固定深度时，手部的面积在五指伸开时最大，成拳形面积最小（约为手掌去除掉五指时的面积）；此时，以掌心为圆心，设置一个合适的阈值圆，则指尖点均分布在圆外。阈值圆指尖点提取法中，掌心位置和阈值圆半径的选取将决定指尖点提取的准确性。

1)掌心计算

前述由 Kinect 手掌跟踪算法直接获取的手掌心三维坐标 (x_H,y_H,z_H) 是包括五指的，无法满足阈值圆指尖点提取法的应用需求。因而，提出一种基于距离变换的掌心精确计算方法，算法流程如下：

a)将手掌图像二值化，手掌设为白色，其他区域黑色，如图 6(a)所示；

b)遍历图像，计算非零像素（白）到最近零像素的距离，图 6(b)中颜色深浅反映该距离大小，离零像素点越远处越亮；

c)设置阈值，将距离小于阈值的像素置为黑，大于阈值则不变，得到图像 6(c)；

d)求处理后图像中心 p，为所有非零像素点图像矩阵位置 x,y 的均值。图 6(d) (e)中分别显示不同手势时掌心位置，该位置将不受手指伸缩状态影响。

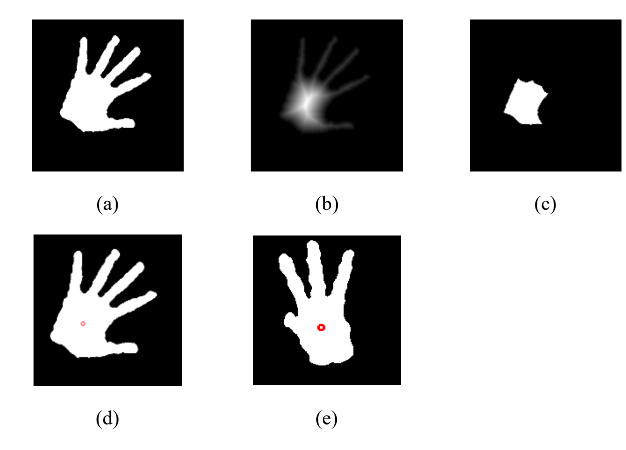


图 6　通过距离变换找到掌心

Fig.6　Find the palm by distance transformation

2)阈值动态调整算法

(1)阈值调整原理

相机成像过程中，物体（长度 L）在图像中的投影长度 l 与物体到相机的距离 $depth$ 成反比，即

$$\frac{l}{L}=\frac{f}{depth} \tag{8}$$

其中: f 为深度相机焦距，$depth$ 表示深度，即手掌距离传感器的距离。

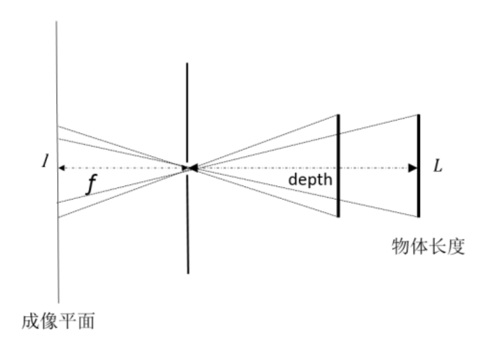


图 7　相机成像原理

Fig.7　Camera imaging principles

对于手掌轮廓图像中阈值圆 r 与实际衡量手掌的圆 R 关系由上述原理可得：

$$r=\frac{fR}{depth} \tag{9}$$

考虑到传感器误差等因素存在，设置缩放误差 λ 和偏移误差系数 β，因此

$$r=\frac{\lambda fR}{depth}+\beta=\frac{\alpha}{depth}+\beta \tag{10}$$

其中：$\alpha=\lambda fR$。

则手掌成像大小与手掌到相机的距离有关。因此，阈值圆法中的阈值半径 r 随着距离 depth 动态调整遵循式(10)。

(2)系数 α、β 回归分析

为了确定式(9)中的系数 α、β，通过实验方法，由 Kinect 采集若干组 r-depth 对应数据，由回归分析获取 α、β 的数值。

Kinect 检测的深度范围是 0.5～4 m 之间，当距离太远时手掌面积太小，受噪声干扰较大，故实验主要是取区间 540~800 mm 的距离。

以深度 540 mm 为例，由采集的五指轮廓图，计算质心点 $p=(px,py)$。由轮廓多边形提取和凸包计算后得到凸包点序列，记作：$\{p_1,p_2,p_3,\dots,p_n\}$。计算质心到所有凸包点的距离的均值：

$$M=\frac{\sum_{i=1}^{n}\left(\sqrt{(p_ix-px)^2+(p_iy-py)^2}\right)}{n} \tag{11}$$

令阈值圆的半径 $r=0.75M$。

测量不同深度 depth 下，在五指伸直时，手掌阈值圆的半径 r，得到样本数据如表 2 所示。

表 2　深度 depth 与手掌阈值圆半径 r 的对应关系

Table 2　Correspondence between depth and radius r of palm threshold circle

depth/mm	540	560	580	600	620	…	720	740	760	780	800
r	84	80	79	75	74	…	63	61	57	60	58

将表 2 数据代入式（8）中，通过线性回归可以获得 α 、β 数值。回归分析结果如图 8 所示，则 $\alpha=45236$，$\beta=0.4838$。

在不同深度条件下，阈值圆半径根据式(9)动态调整，以消除手掌距离相机位置不同对阈值圆分类法的影响，较好的剔除无关点，提高指尖点检测的准确性，减小对手掌运动范围的限

制。

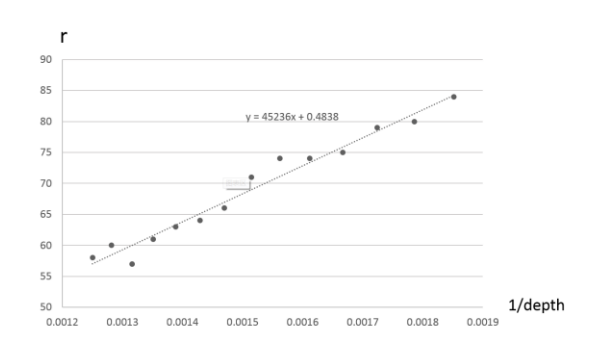


图 8 半径与深度的关系

Fig.8　Relationship between radius and depth

2.2.4 阈值圆指尖点分类

在不同深度、不同手势的情况下，通过可以动态调整的阈值圆，去除阈值圆内的无关点，保留阈值圆外的指尖点。该方法能够得到较好的分类效果。如图 9 所示，大圈为动态阈值圆，其外部标记点是从候选指尖点中分离出来的有效指尖点。

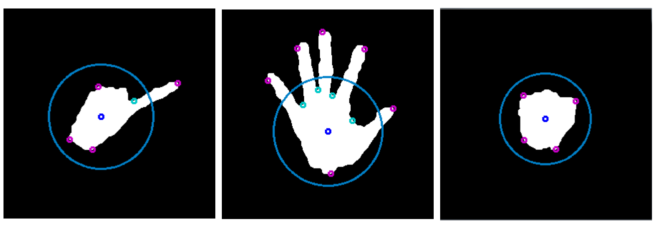


图 9 不同深度、不同手势下的阈值圆和指尖点提取

Fig.9　Threshold cycle and fingertip points extraction under different depths,different gestures

经过几何不变矩特征量计算和指尖点提取，得到手势特征 $X=\{I_1^{'},I_2^{'},I_3^{'},I_4^{'},a\}$，其中，$I_1^{'},I_2^{'},I_3^{'},I_4^{'}$ 是几何不变矩的前四个特征量，a 是指尖点数。手势识别在此基础上展开。

3　K 最近邻手势识别算法

通过手势特征提取得到特征向量 $X=\{I_1^{'},I_2^{'},I_3^{'},I_4^{'},a\}$，在此基础上，利用分类算法对手势进行分类，以完成手势识别。由于需要识别的手势种类较多，且需要较高的实时性，所以采用 K 最近邻算法（KNN）作为分类方法。KNN 分类器无须训练，无须估计参数，在多分类问题中具有较好的分类效果，是一种简单高效的分类算法。

K 最近邻算法的核心思想是构建一个样本集，包括若干组标签不同的样本子集，每个样本子集是由特征向量相近的样本组成。在分类时，输入测试样本，找到样本集中前 K 个最相似（距离最近）的样本，其中出现次数最多的标签即为该测试样本的类别。

在手势识别中，以手掌几何不变矩和指尖点个数组成特征向量 $X=\{I_1^{'},I_2^{'},I_3^{'},I_4^{'},a\}$，作为分类依据。手势识别时采用欧氏距离 d，度量样本之间的相似性。

$$d=\sqrt{\sum_{i=1}^{n}(x_{1i}-x_{2i})^2} \qquad (12)$$

其中：$X_1=[x_{11},x_{12},x_{13},x_{14},x_{15}]$ 为测试样本特征向量，$X_2=[x_{21},x_{22},x_{23},x_{24},x_{25}]$ 为样本集特征向量，n=5 为手势特征向量的维度。

在 KNN 算法中，参数 k 的取值对分类准确率有较大影响。k 值较大时，可以减少干扰数据的影响，但缺点是近似误差会增大；当 k 值较小，会导致过度拟合。通常 k 值设置为 $\sqrt{m}$，m 为样本数量。

为防止某个特征过大或者过小，引起分类过程中各特征所起作用严重失衡，需要对数据进行归一化；同时，归一化有利于提高算法速度。在此，采用线性归一化方法。

$$y=\frac{x-minVal}{maxVal-minVal} \qquad (13)$$

其中：$maxVal$、$minVal$ 分别代表全部样本中该维数据的最大值和最小值。x 为待归一化的数值，y 即为归一化结果，其值介于 0~1 之间。

本文手势识别算法中，样本集由标签为“手势 1”到“手势 6”的六个子集组成，每个子集由事先采集的 15 个同类手势组成。KNN 手势识别算法中，通过 KNN 分类器，获取待测手势的标签号，完成手势识别。

4　算法流程

基于复合特征和动态阈值圆法的手势识别算法的流程如图 10 所示。

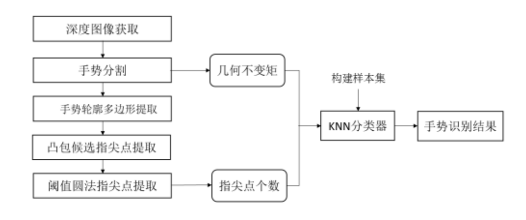


图 10 手势识别流程图

5　手势识别实验

为验证基于复合特征和动态阈值圆法的手势识别算法的有效性，进行相关实验。实验基于 PC 平台与 Kinect 传感器。PC 具有 2.9 GHz 的 CPU 与 8 GB 的 RAM；Kinect 的采样图片大小为 320*240，帧率 30 fps，能达到实时识别的硬件要求。利用 OpenNI 和 OpenCV 开源库在 Visual Studio 搭建实验平台。

实验过程中，三名不同的实验者随机作出编号为 1-6 的 6 种手势，其中编号 1-5 的手势分别为手指伸出 1 到 5 指，编号 6 即为手掌为拳形的手势。手掌位置和手掌离 Kinect 的距离也随机变换，共计采集 240 组手势数据，手势识别结果如表 3 所示。

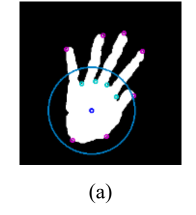
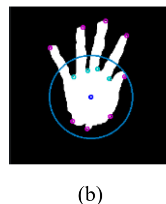
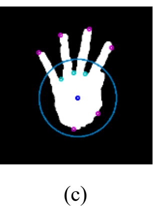
从表 3 中可以看出，在动态调节阈值圆的情况下，对于不同手势，平均识别率达到 96.25%以上。在实验过程中，其识别响应时间平均为 0.65 s，算法具有较高的实时性。

手势 5 的识别率较其他手势稍差，仅为 92.50%。通过分析及实验验证，这一现象是由于大拇指较其他手指短，当其过于靠近食指时，易于造成错误识别。如图 11 所示，图 11(a)可正确识别，图 11(b)大拇指过于靠近手掌，识别已出现偏差，而图 11(c) 与手势 4 相似，导致错误识别。

表 3　势识别实验结果

Table 3　Gesture recognition experiment results

手势编号	样本数	识别正确数	识别率
1	40	40	100%
2	40	39	97.5%
3	40	38	95.0%
4	40	38	95.0%
5	40	37	92.5%
6	40	39	97.5%
总计	240	231	96.25%

(a)　(b)　(c)

图 11 大拇指位置对手势 5 识别的影响

Fig.11　Effect of thumb position for gesture 5 recognition

针对手势 5 这一问题，重复采集数据，要求大拇指不得过于靠近食指，对采集的 30 组手势数据进行识别，识别正确的样本数为 29，识别率为 96.67%。

KNN 算法的主要参数是 K 值，K 值过大或过小均可能会对结果造成影响。前文中采集 90 组样本集，每个手势各 15 个，由于 K 值的选择通常是奇数，因此 K 值从 3 开始到 13，以相同的测试样本，改变 K 值，重新统计手势识别的准确率。如图 12 所示。

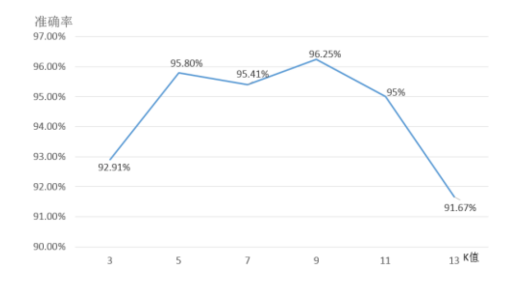


图 12　k 值对准确率的影响

Fig.12　Effect of k value on accuracy

从图中可已看出 k 值过大或者过小对准确率都有不同程度的下降，当 k 介于 5~11 时，准确率均能达到 95%以上。

为了与 KNN 算法做对比，本文引入 BP 神经网络和支持向量机（SVM）分类器对手势进行识别，其训练样本集与前文 KNN 算法样本集的相同，测试样本集也相同。

从表中可以看出 KNN 算法的识别率最高。其根本原因在于 BP 神经网络算法和 SVM 算法都是需要对样本进行训练，对样本的数量和质量要求较高，样本数量过少会导致训练的模型泛化能力较差，识别效果不佳。因此本文通过增加训练样本数量将总样本数量提高到 180 组，SVM 分类的识别率提升到了 96.67%，BP 神经网络的提高到了 95.0%。与 KNN 算法相比，SVM 分类器和 BP 神经网络均要对样本进行训练，会花费较长时间，而 KNN 算法无须训练，算法的流程也十分简单，因此花费时间与前两种算法相比较少，能满足本文手势实时识别的要求。

综上所述，对于不同人的手部特征，均有较好地识别效果。实验证明了基于复合特征和动态阈值圆法的手势识别算法的有效性和实时性。

表 4　不同分类算法的手势识别结果

分类算法	识别率/%
SVM 分类器	90.83
BP 神经网络	83.33
KNN	96.25

6　结束语

本文利用深度图像，通过深度阈值分割手部区域。引入图像的几何不变矩作为分类的特征，与指尖点特征一起构成复合手势特征。提取指尖点时，动态调整阈值圆以除去噪点。最后以图像的矩和指尖点为分类特征，用最近邻算法（KNN）进行分类，实现手势识别。基于复合特征和动态阈值圆法的手势识别算法能够有效进行手势识别，识别率高、实时性好。

参考文献：

[1] Sturman D J, Zeltzer D, Pieper S. Hands-on interaction with virtual environments [C]// Proc of ACM GRAPH Symposium on User Interface Software and Technology. New York: ACM Press, 1989: 19-24.

[2] 李勇，高文，姚鸿勋. 基于颜色手套的中国手指语字母的动静态识别 [J]. 计算机工程与应用, 2002, 38 (17): 55-58. (Li Yong, Gao Wen, Yao Hongxun. Dynamic and Static Recognition of Chinese Finger Letters Based on Color Gloves [J]. Computer Engineering and Applications, 2002, 38 (17): 55-58.)

[3] 任海兵，祝远新，徐光，等. 基于视觉手势识别的研究综述 [J]. 电子学报, 2000, 28 (2): 118-121. (Ren Haibing, Zhu Yuanxin, Xu Guangzheng *et al.* Research based on visual gesture recognition: a review [J]. Chinese Journal of Electronics, 2000, 28 (2): 118-121.)

[4] 邱迪. 基于 HSV 与 YCrCb 颜色空间进行肤色检测的研究 [J]. 电脑编程技巧与维护, 2012 (10): 74-75. (Qiu Di. Research on Skin color detection based on hsv and ycrcb color space [J]. Computer Programming Skills and Maintenance, 2012 (10): 74-75.)

[5] 李擎，唐欢，迟健男，等. 基于改进最大类间方差法的手势分割方法研究 [J]. 自动化学报, 2017, 43 (4): 528-537. (Li Qing, Tang Huan, Chi Jianan, *et al.* Research on gesture segmentation based on improved maximum variance between classes [J]. Automation Journal, 2017, 43 (4):



528-537.)

[6] 许凯，王敏. 基于手轮廓的深度图像手势识别方法 [J]. 计算机工程与科学, 2014, 36 (5): 941-946. (Xu Kai, Wang Min. Gesture recognition based on hand contour in depth image [J]. Computer Engineering & Science, 2014, 36 (5): 941-946.)

[7] 李瑞峰，曹雏清，王丽. 基于深度图像和表观特征的手势识别 [J]. 华中科技大学学报：自然科学版, 2011, 39 (s2): 88-91. (Li Ruifeng, Cao Xiaoqing, Wang Li. Gesture recognition based on depth images and appearance features [J]. Journal of Huazhong University of Science and Technology: Nature Science, 2011, 39 (s2): 88-91.)

[8] Yoruk E, Konukoglu E, Sankur B, *et al.* Shape-based hand recognition [J]. IEEE Trans on Image Processing, 2006, 15 (7): 1803.

[9] 谈家谱，徐文胜. 基于 Kinect 的指尖检测与手势识别方法 [J]. 计算机应用, 2015, 35 (6): 1795-1800. (Tan Jiapu, Xu Wensheng. Fingertip detection and hand gesture recognition method based on Kinect [J]. Journal of Computer Applications, 2015, 35 (6): 1795-1800.)

[10] 高晨，张亚军. 基于 Kinect 深度图像的指尖检测与手势识别 [J]. 计算机系统应用, 2017, 26 (4): 192-197. (Gao Chen, Zhang Yajun. Fingertip detection and gesture recognition based on Kinect depth image [J]. Computer Systems & Applications, 2017, 26 (4): 192-197.)

[11] 杨学文，冯志全，黄忠柱，等. 结合手势主方向和类-Hausdorff 距离的手势识别 [J]. 计算机辅助设计与图形学学报, 2016, 28 (1): 75-81. (Yang Xuewen, Feng Zhiquan, Huang Zhongzhu, *et al.* Gesture recognition combined with gesture main direction and class-Hausdorff distance [J]. Journal of Computer-Aided Design & Computer Graphics, 2016, 28 (1): 75-81.)

[12] Oden C, Ercil A, Buke B. Combining implicit polynomials and geometric features for hand recognition [J]. Pattern Recognition Letters, 2003, 24 (13): 2145-2152.

[13] 崔家礼，解威，王一丁，等. 基于自适应手指分割与判别的静态手势识别 [J]. 计算机应用与软件, 2016, 33 (10): 181-186. (Cui Jiali, Jie Wei, Wang Yiding, *et al.* Static gesture recognition based on adaptive finger segmentation and discrimination [J]. Computer Applications and Software, 2016, 33 (10): 181-186.)

[14] Hu M K. Visual pattern recognition by moment invariants [J]. Information Theory Ire Transactions on, 1962, 8 (2): 179-187.

[15] 王艳，徐诗艺，谌海云. 基于特征距离加权的手势识别 [J]. 计算机科学, 2017, 44 (s1): 220-223. (Wang Yan, Xu Shiyi, Yan Haiyun. Gesture Recognition Based on Feature Distance Weighting [J]. Computer Science, 2017, 44 (s1): 220-223.)